

FIG - 1  
PRIOR ART

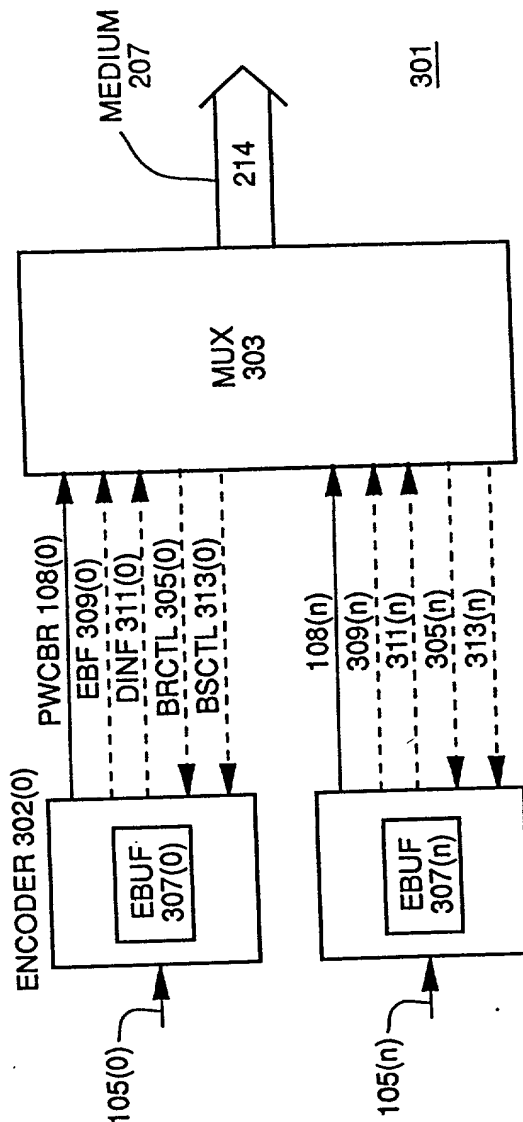
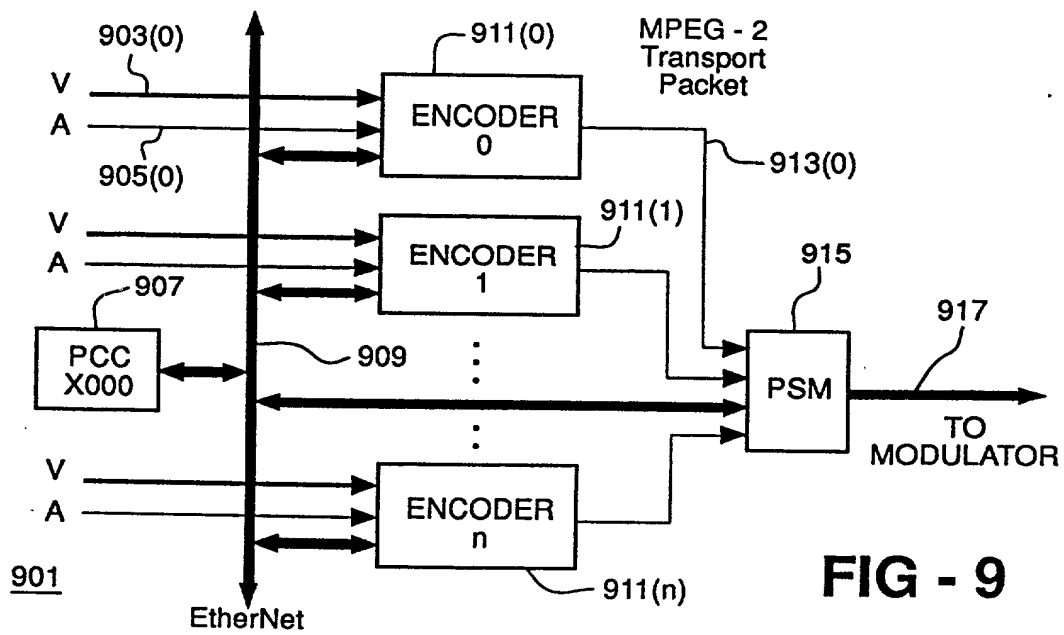
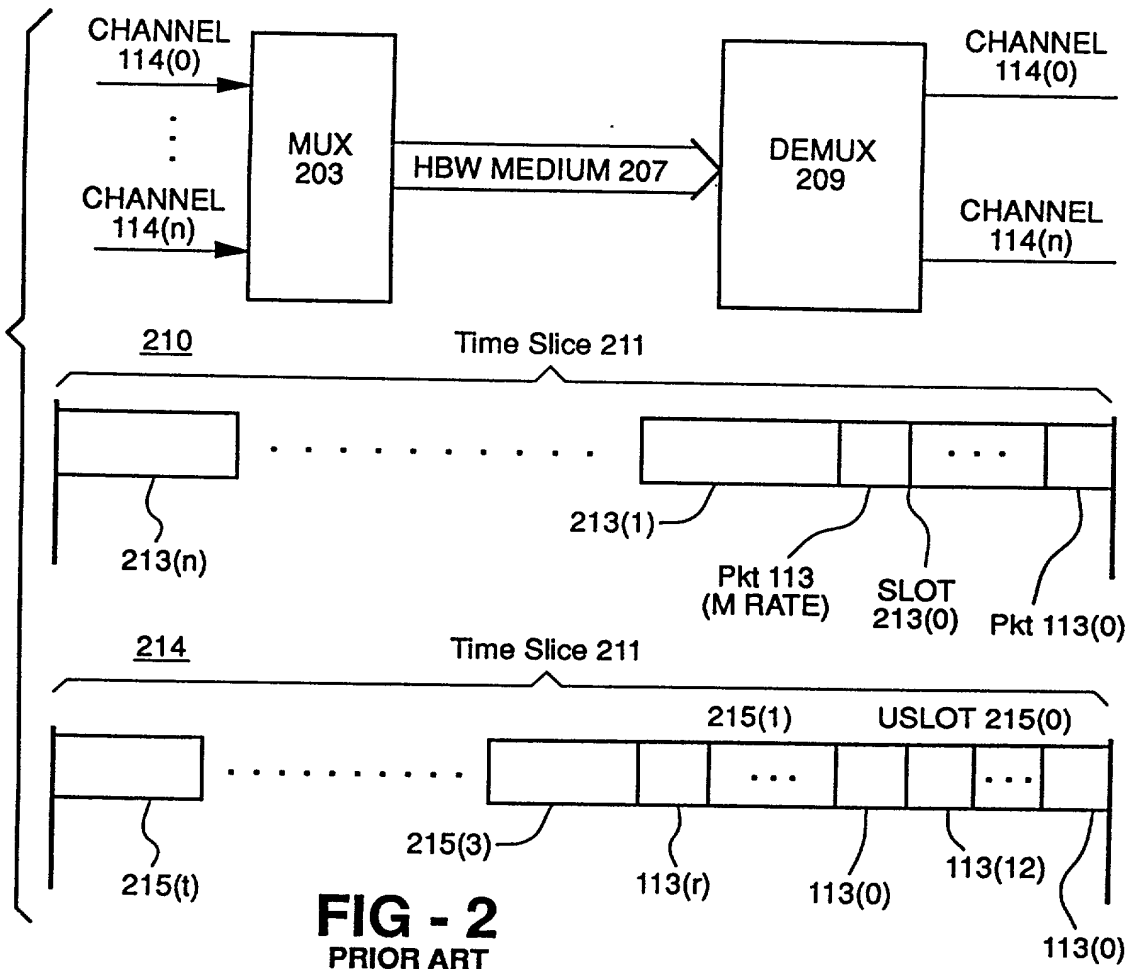
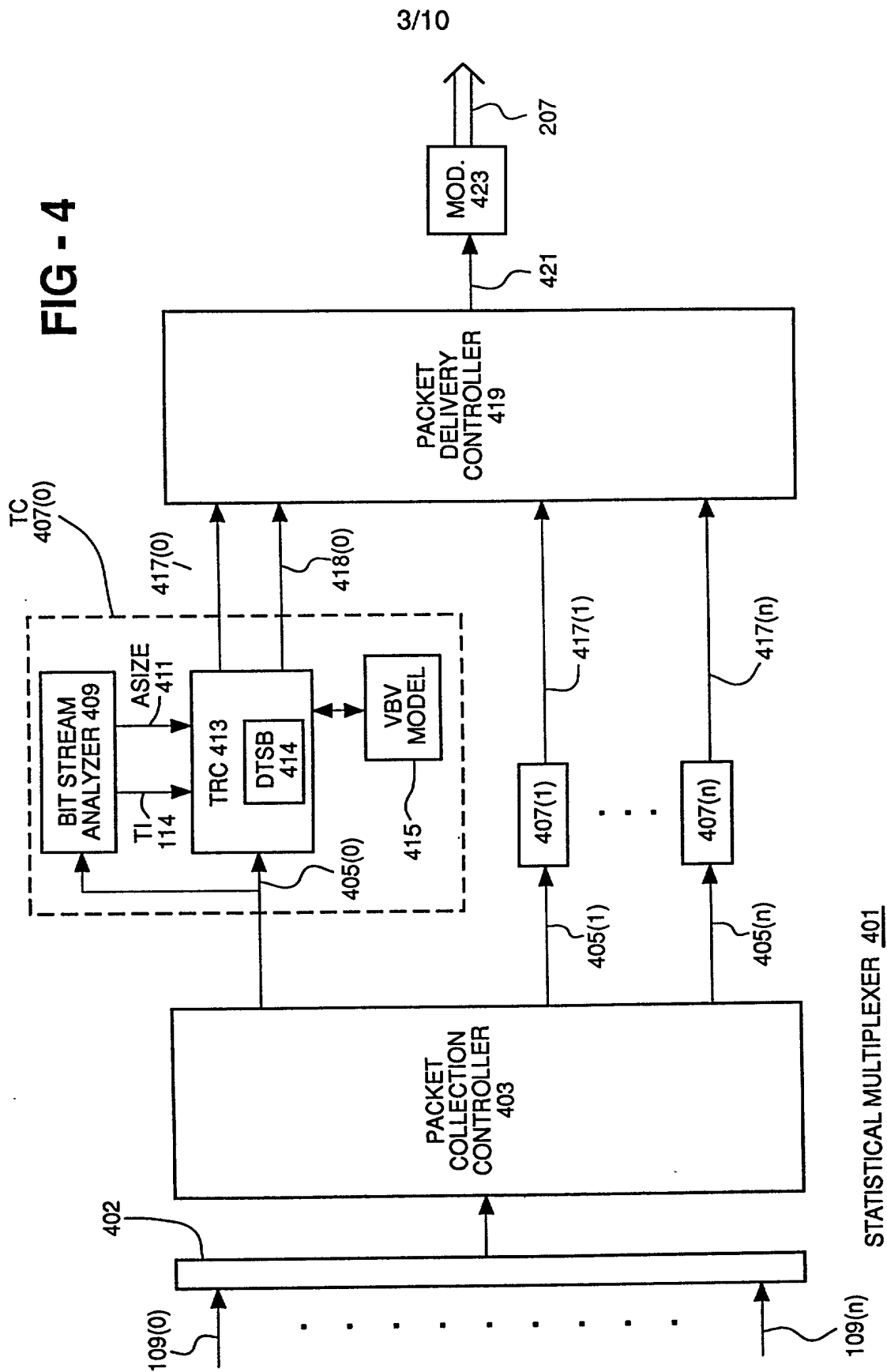


FIG - 3  
PRIOR ART





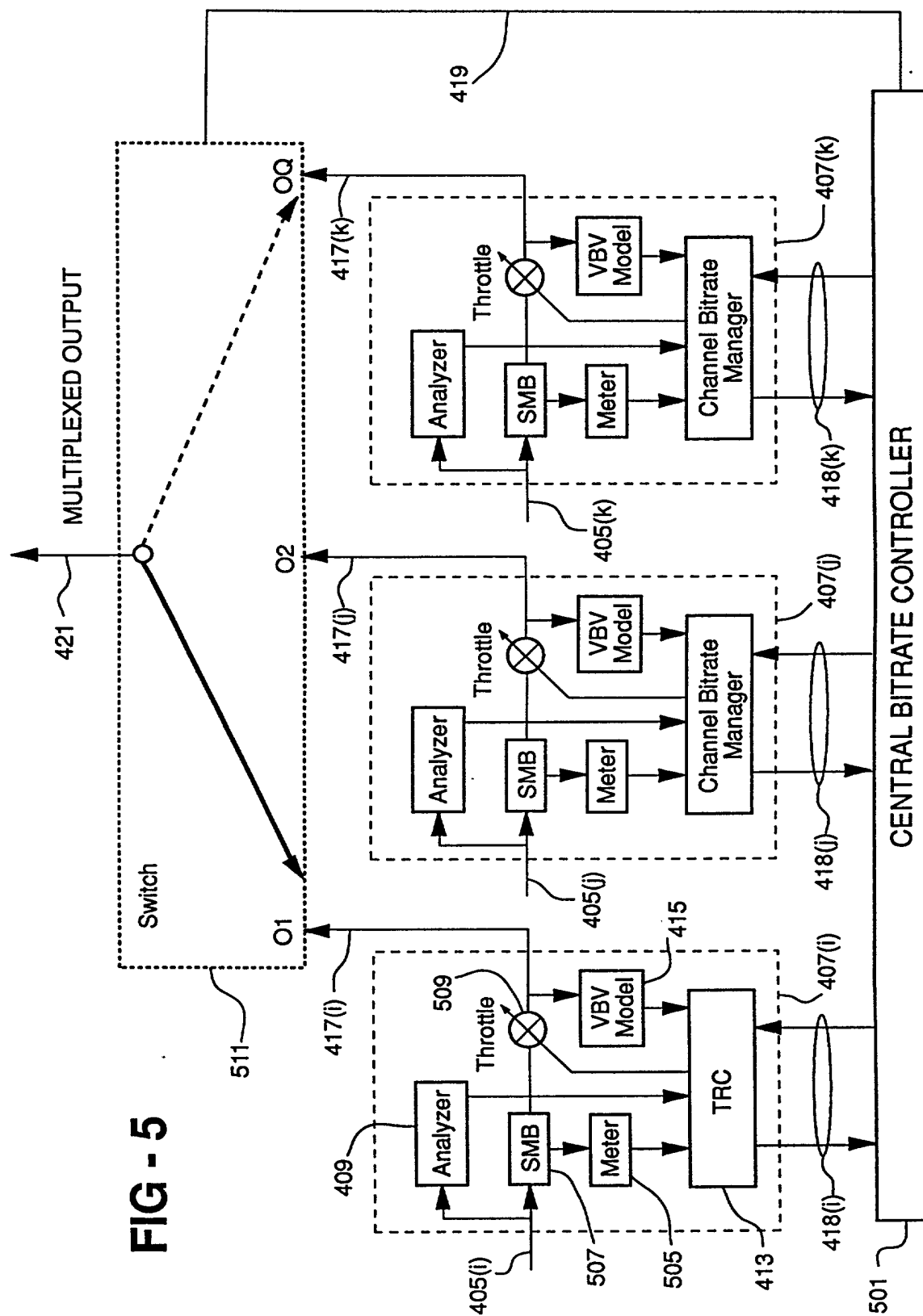


FIG - 6

```

/* Initialization of parameters */
q = 0;
r = 0;
pic_cnt_VBV = 0;
pic_residual_bits = packet_cnt(0)*188; /* Initial condition when no bit has left SMB */
Fvbv(0) = 0;

for (m = 1; ; m++) {
    /* q is the index of coded picture that is leaving SMB */
    /* r is the index of coded picture that is leaving VB */
    /* picture counter in VB buffer */
    /* Decoder's VB fullness is empty at the beginning */
    /* m is the index of each channel transmission time Tc */
    /* update the pic_residual_bits parameter in SMB */
    /* Rm is the bitrate for the mth Tc period */
    pic_residual_bits -= R(m)*Tc;

    /* VB fullness update */
    if (t < DTS(r)) {
        /* The picture r has not been removed from VB. Here the t is the real
        time derived from PCR */
        Fvbv(m) += R(m) * Tc;
    } else {
        /* picture r has been completely removed from VB */
        /* The number of coded picture in VB should be reduced by 1 */
        pic_cnt_VBV--;
        Fvbv(m) += R(m) * Tc - packet_cnt(r)*188;
        r = (r + 1) % NPPmax;
    }
    /* Note: the VB fullness calculated in this way has the TS and PES overhead.
    It means the actual fullness of decoder's bit buffer is less than VB fullness.
    We don't intend to exclude such overhead since it is rather tedious. Instead
    we use this overhead as an extra small buffer at the top of decoder's bit
    buffer to guarantee there is no overflow */

    if (pic_residual_bits < 0) {
        /* last coded picture has been completely moved from
        SMB to VB */

        /* Update the maximum DTS value and picture counter information in VB */
        DTS_Vmax = DTS(q);
        /* The maximum DTS value in VB */
        pic_cnt_VBV++;
        /* Number of complete coded picture in VB */

        /* Update parameters for SMB */
        q = (q + 1) % NPP;
        /* next q value q has the same value range as NPP */
        pic_residual_bits += packet_cnt(q)*188; /* add in the bits amount for the new picture */
    }
}

```

FIG - 7

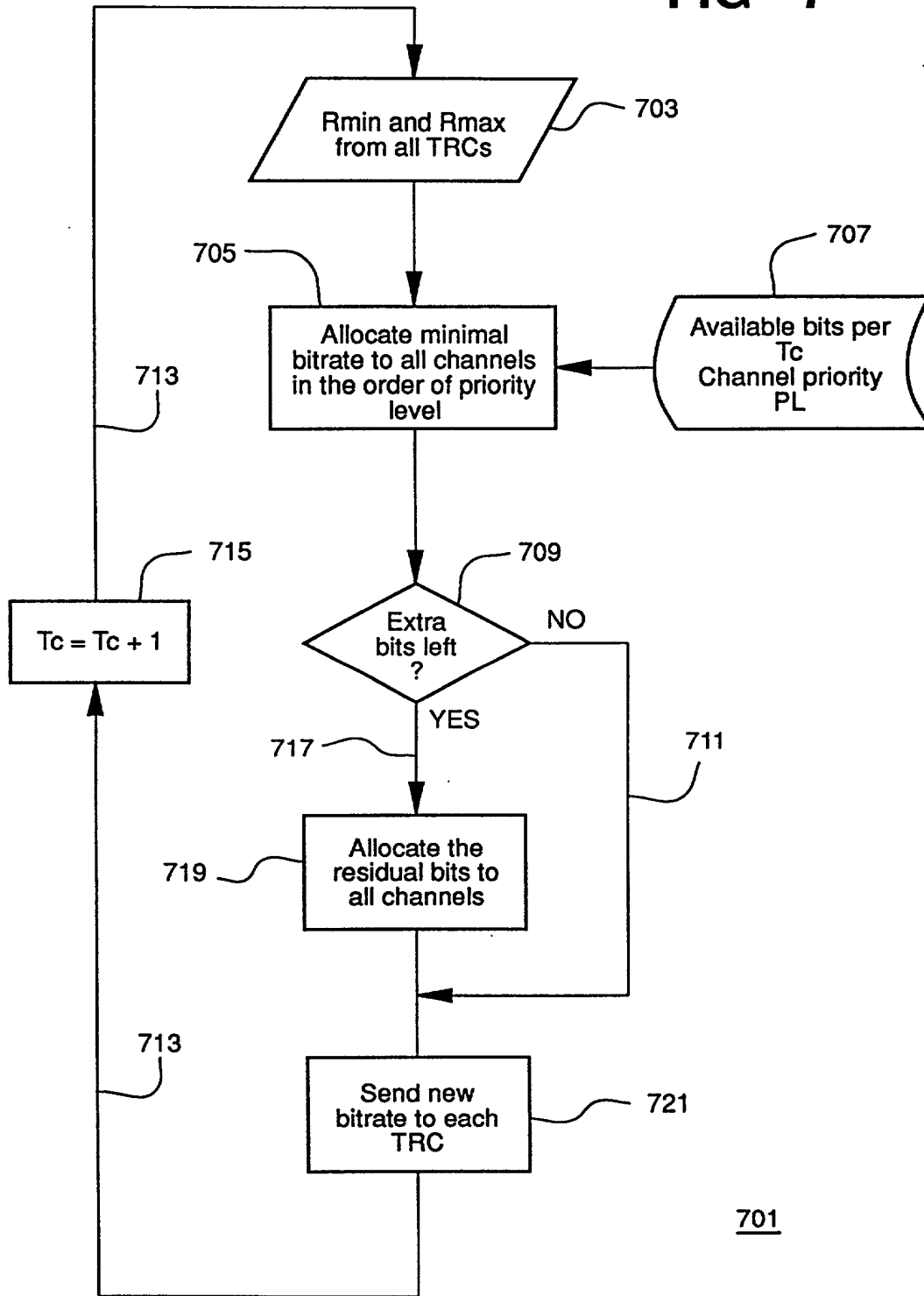
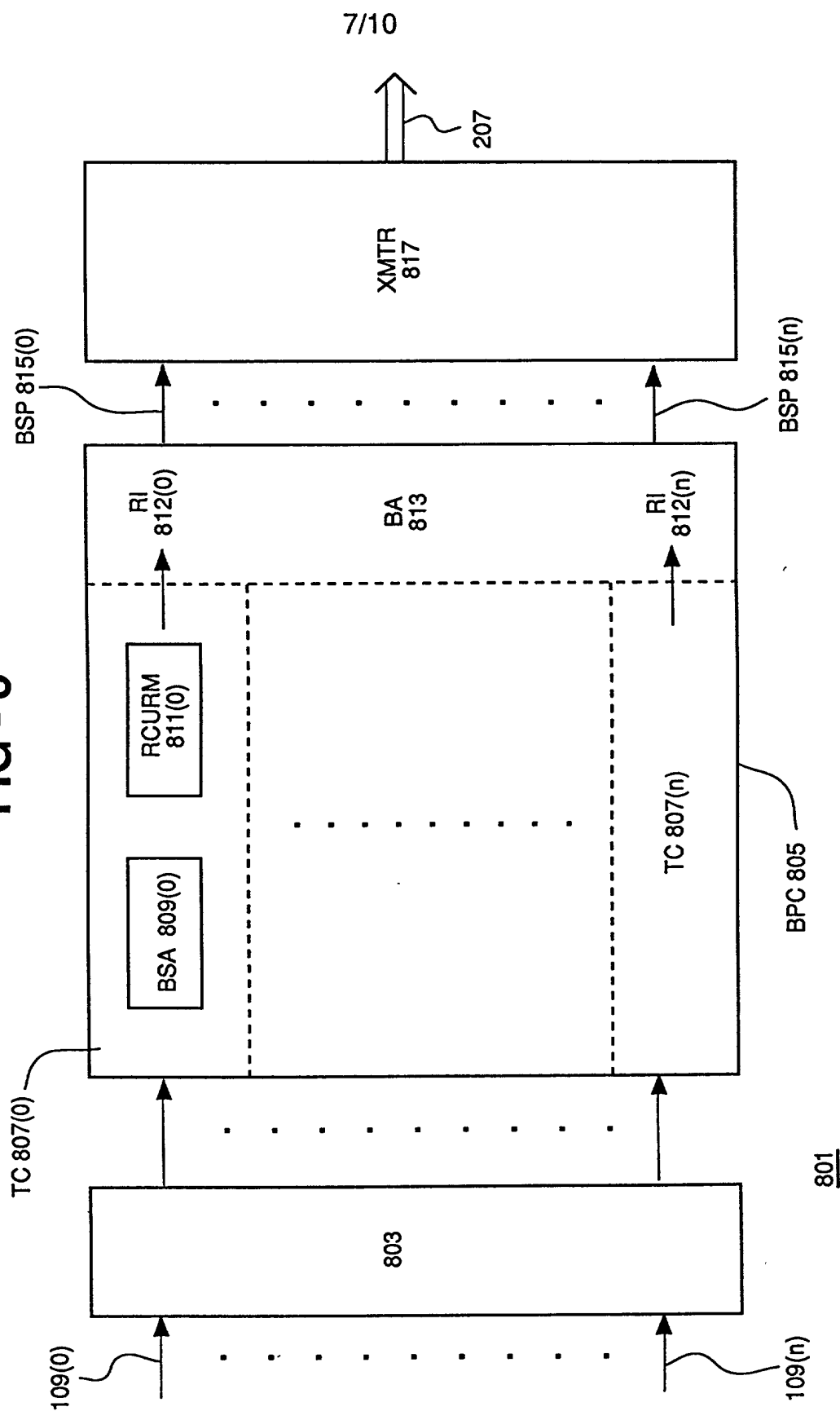
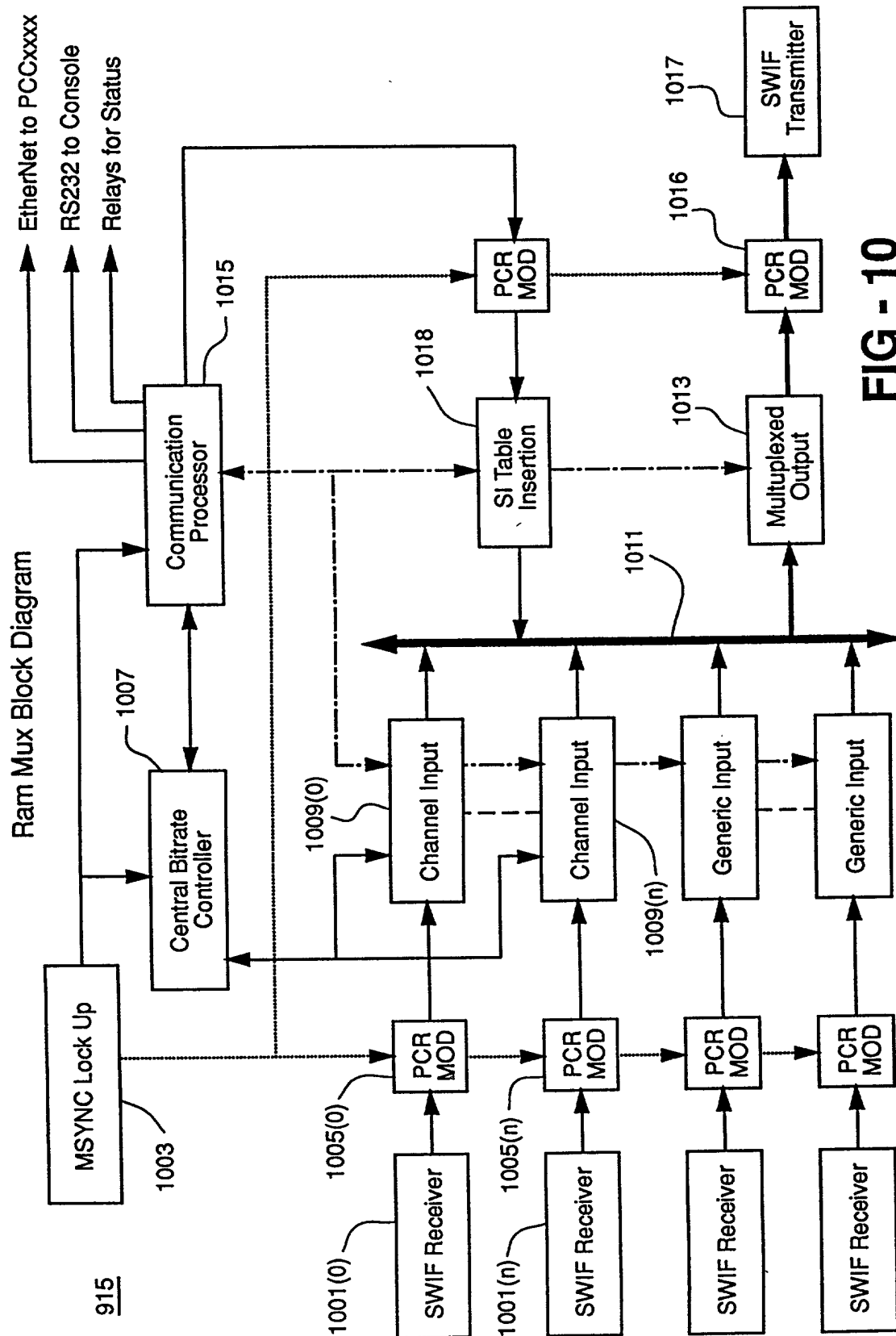


FIG - 8





**FIG - 10**



**FIG - 11**

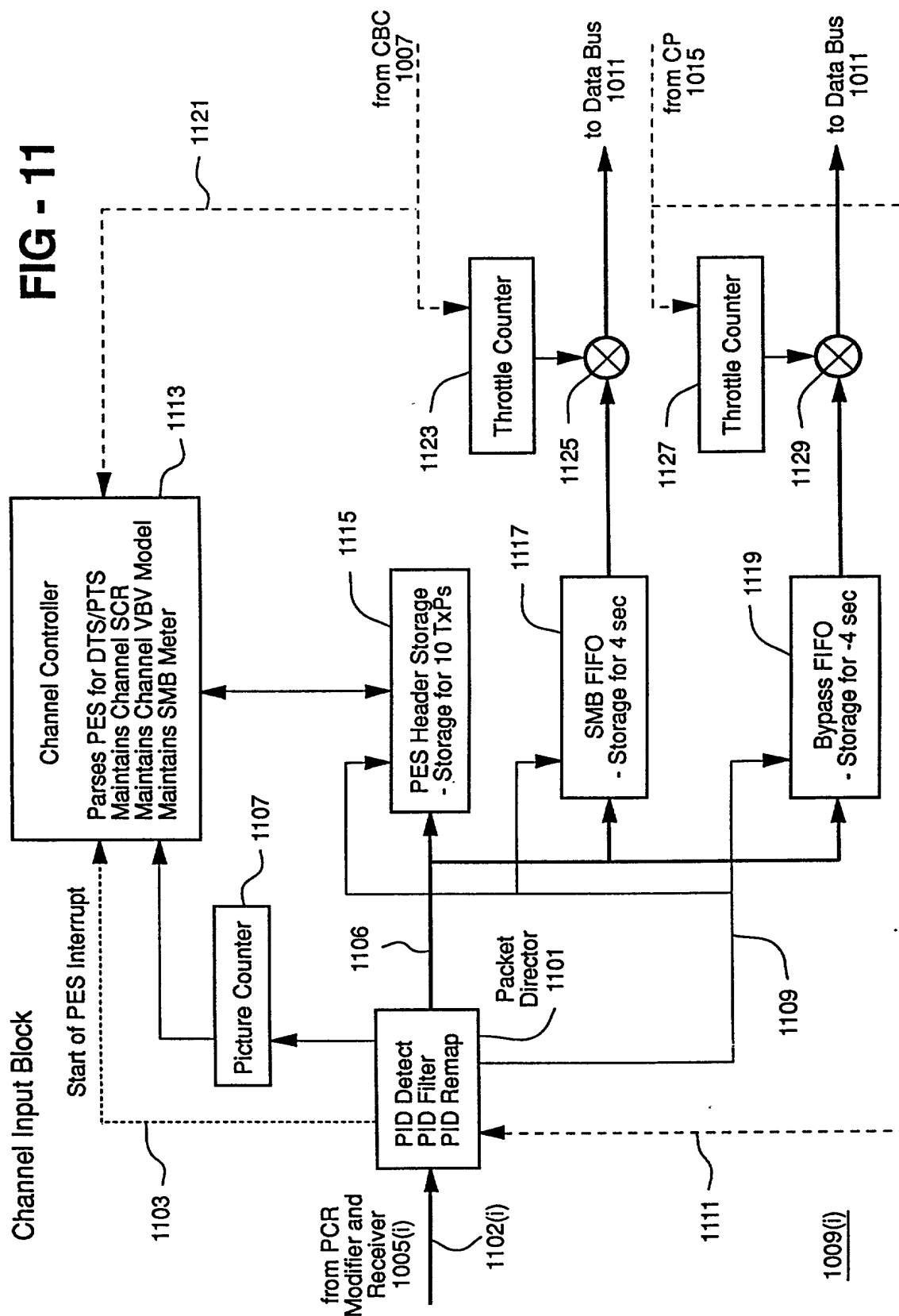


FIG - 12

